	Type	Hits	Search Text	DBs
1	IS&R	53	(("5647036") or ("5630004") or ("5586206") or ("5581642") or ("5544268") or ("5504772") or ("5491762") or ("5488681") or ("6167169") or ("5664032") or ("5652817") or ("6141465") or ("6118908") or ("6078704") or ("5978524") or ("5912997") or ("5911018") or ("5887089") or ("5852688") or ("5835458") or ("5781670") or ("5732177") or ("5724463") or ("5703710")).PN.	USPAT; EPO; JPO; DERWENT; IBM_TDB
2	IS&R	4356	((372/38.04) or (372/29.012) or (372/50) or (257/99) or (257/207) or (257/775) or (372/109)).CCLS.	USPAT; EPO; JPO; DERWENT; IBM_TDB
3	BRS	412797	circuit and current and power	USPAT; EPO; JPO; DERWENT; IBM_TDB
4	BRS	21981	(plurality multiple) near3 electric$4 with conduct$5	USPAT; EPO; JPO; DERWENT; IBM_TDB
5	BRS	29	((("372/38.04") or ("372/29.012") or ("372/50") or ("257/99") or ("257/207") or ("257/775") or ("372/109")).CCLS.) and (circuit and current and power) and ((plurality multiple) near3 electric$4 with conduct$5)	USPAT; EPO; JPO; DERWENT; IBM_TDB
6	BRS	71	((("372/38.04") or ("372/29.012") or ("372/50") or ("257/99") or ("257/207") or ("257/775") or ("372/109")).CCLS.) and ((plurality multiple) near3 electric$4 with conduct$5)	USPAT; EPO; JPO; DERWENT; IBM_TDB
7	BRS	42	((((("372/38.04") or ("372/29.012") or ("372/50") or ("257/99") or ("257/207") or ("257/775") or ("372/109")).CCLS.) and ((plurality multiple) near3 electric$4 with conduct$5)) not ((((("372/38.04") or ("372/29.012") or ("372/50") or ("257/99") or ("257/207") or ("257/775") or ("372/109")).CCLS.) and (circuit and current and power) and ((plurality multiple) near3 electric$4 with conduct$5))	USPAT; EPO; JPO; DERWENT; IBM_TDB

	Type	Hits	Search Text	DBs
8	BRS	4978	(circuit and current and power) and ((plurality multiple) near3 electric$4 with conduct$5)	USPAT; EPO; JPO; DERWENT; IBM_TDB
9	BRS	4949	((circuit and current and power) and ((plurality multiple) near3 electric$4 with conduct$5)) not (((((("372/38.04") or ("372/29.012") or ("372/50") or ("257/99") or ("257/207") or ("257/775") or ("372/109")).CCLS.) and (circuit and current and power) and ((plurality multiple) near3 electric$4 with conduct$5)) (((((("372/38.04") or ("372/29.012") or ("372/50") or ("257/99") or ("257/207") or ("257/775") or ("372/109")).CCLS.) and ((plurality multiple) near3 electric$4 with conduct$5)) not ((((("372/38.04") or ("372/29.012") or ("372/50") or ("257/99") or ("257/207") or ("257/775") or ("372/109")).CCLS.) and (circuit and current and power) and ((plurality multiple) near3 electric$4 with conduct$5))))	USPAT; EPO; JPO; DERWENT; IBM_TDB
10	BRS	18	("H tree" "H-tree") adj network	USPAT; EPO; JPO; DERWENT; IBM_TDB
11	BRS	0	("X tree" "X-tree") adj network	USPAT; EPO; JPO; DERWENT; IBM_TDB
12	BRS	1	("X shaped" "X-shaped") adj network	USPAT; EPO; JPO; DERWENT; IBM_TDB
13	IS&R	1	("5309001").PN.	USPAT
14	BRS	24272	level and (power NEAR5 distribut$6)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB
15	BRS	19	(level and (power NEAR5 distribut$6)) and (("H tree" "H-tree"))	USPAT; EPO; JPO; DERWENT; IBM_TDB
16	BRS	169	("H tree" "H-tree")	USPAT; EPO; JPO; DERWENT; IBM_TDB
17	BRS	1320402	clock distribution	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB

	Type	Hits	Search Text	DBs
18	BRS	2970	clock adj distribution	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB
19	BRS	1150	clock adj distribution and electric$2	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB
20	BRS	47	("3473160" | "4197555" | "4706216" | "4910417" | "4933738" | "4937475" | "5132571" | "5157627" | "5179534" | "5191241" | "5260597" | "5341041" | "5343403" | "5357560" | "5377225" | "5404033" | "5420544" | "5432719" | "5488316" | "5495486" | "5512765" | "5526278" | "5550839" | "5565695" | "5566123" | "5581098" | "5581200" | "5585602" | "5631577" | "5781031" | "5781033" | "5801547" | "5825202" | "5825203" | "5861325" | "5920789" | "5940727" | "5959466" | "5999016" | "6054872" | "6066960" | "6957380" | "6130554" | "6150838" | "6169416" | "6252792" | "6353352" | "6433585").PN.	USPAT

Your search matched **193** of **974953** documents.

A maximum of **193** results are displayed, **15** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.
Then click **Search Again.**

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

106 **A 10 GHz hybrid optical/electrical clock distribution network for gigascale integration**
Mule, A.V.; Schultz, S.M.; Gaylord, T.K.; Meindl, J.D.;
Lasers and Electro-Optics Society 1999 12th Annual Meeting. LEOS '99. IEEE , Volume: 2 , 8-11 Nov. 1999
Page(s): 627 -628 vol.2

[Abstract] [PDF Full-Text (180 KB)] **IEEE CNF**

107 **An exact tree-based structural technology mapping algorithm for configurable logic blocks in FPGAs**
Lee, K.K.; Wong, D.F.;
Computer Design, 1999. (ICCD '99) International Conference on , 10-13 Oct. 1999
Page(s): 216 -221

[Abstract] [PDF Full-Text (144 KB)] **IEEE CNF**

108 **Model combination and weight selection criteria for speaker verification**
Farrell, K.R.;
Neural Networks for Signal Processing IX, 1999. Proceedings of the 1999 IEEE Signal Processing Society Workshop , 23-25 Aug. 1999
Page(s): 439 -448

[Abstract] [PDF Full-Text (428 KB)] **IEEE CNF**

109 **The Ultrascalar processor-an asympt tically scalable superscalar micr architecture**
Henry, D.S.; Kuszmaul, B.C.; Viswanath, V.;
Advanced Research in VLSI, 1999. Proceedings. 20th Anniversary Conference on , 21-24 March 1999
Page(s): 256 -273

[Abstract] [PDF Full-Text (228 KB)] **IEEE CNF**

---

110 **Optimal location of high-speed facility in heterogeneous networks**
Touyama, T.; Takahashi, A.; Horiguchi, S.;
Parallel Architectures, Algorithms and Networks, 2000. I-SPAN 2000. Proceedings. International Symposium on , 7-9 Dec. 2000
Page(s): 246 -251

[Abstract] [PDF Full-Text (464 KB)] **IEEE CNF**

---

111 **An optical clock distribution network for gigascale integration**
Mule, A.V.; Schultz, S.M.; Gaylord, T.K.; Meindl, J.D.;
Interconnect Technology Conference, 2000. Proceedings of the IEEE 2000 International , 5-7 June 2000
Page(s): 6 -8

[Abstract] [PDF Full-Text (276 KB)] **IEEE CNF**

---

112 **A clock distribution network for microprocessors**
Restle, P.J.; McNamara, T.G.; Webber, D.A.; Camporese, P.J.; Eng, K.F.; Jenkins, K.A.; Allen, D.H.; Rohn, M.J.; Quaranta, M.P.; Boerstler, D.W.; Alpert, C.J.; Carter, C.A.; Bailey, R.N.; Petrovic, J.G.; Krauter, B.L.; McCredie, B.D.;
VLSI Circuits, 2000. Digest of Technical Papers. 2000 Symposium on , 15-17 June 2000
Page(s): 184 -187

[Abstract] [PDF Full-Text (436 KB)] **IEEE CNF**

---

113 **Steiner-optimal data replication in tree networks with storage costs**
Kalpakis, K.; Dasgupta, K.; Wolfson, O.;
Database Engineering & Applications, 2001 International Symposium on. , 16-18 July 2001
Page(s): 285 -293

[Abstract] [PDF Full-Text (708 KB)] **IEEE CNF**

---

114 **Universal self-stabilizing phase cl ck pr t c l with b unded mem ry**
Nolot, F.; Villain, V.;
Performance, Computing, and Communications, 2001. IEEE International Conference on. , 4-6 April 2001
Page(s): 228 -235

[Abstract] [PDF Full-Text (484 KB)] **IEEE CNF**

---

115 **New efficient and accurate moment matching based model for crosstalk estimation in coupled RC trees**
Qingjian Yu; Kuh, E.S.;
Quality Electronic Design, 2001 International Symposium on , 26-28 March 2001
Page(s): 151 -157

[Abstract] [PDF Full-Text (563 KB)] **IEEE CNF**

---

116 **Information update on neural tree networks**
Gentili, S.;
Image Processing, 2001. Proceedings. 2001 International Conference on , Volume: 1 , 7-10 Oct. 2001
Page(s): 505 -508 vol.1

[Abstract] [PDF Full-Text (320 KB)] **IEEE CNF**

---

117 **Adaptation of data fusion-based speaker verification models**
Farrell, K.R.;
Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on , Volume: 2 , 26-29 May 2002
Page(s): II-851 -II-854 vol.2

[Abstract] [PDF Full-Text (426 KB)] **IEEE CNF**

---

118 **Demonstration of a broadband wireless backbone using ultra-dense WDM technology**
Jie Lin; Fow-Sen Choa;
Lasers and Electro-Optics, 2002. CLEO '02. Technical Digest. Summaries of Papers Presented at the , 19-24 May 2002
Page(s): 276 -277 vol.1

[Abstract] [PDF Full-Text (385 KB)] **IEEE CNF**

119 **High speed ptical netw rk f r asynchr n us multiuser access applying peri dic spectral c ding f br adband s urces**

Pfeiffer, T.; Deppisch, B.; Kaiser, M.; Heidemann, R.;
Electronics Letters , Volume: 33 Issue: 25 , 4 Dec. 1997
Page(s): 2141 -2142

[Abstract] [PDF Full-Text (260 KB)] **IEE JNL**

120 **Assigning a Minimal Multiple-Center Under a Given Reliability Index in Tree Networks**
Trstensky, D.;
Communications, IEEE Transactions on [legacy, pre - 1988] ,
Volume: 27 Issue: 3 , Mar 1979
Page(s): 638 -639

[Abstract] [PDF Full-Text (152 KB)] **IEEE JNL**

[Prev] 1 2 3 4 5 6 7 8 9 10 11 12 13 [Next]

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search
Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting
No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

Your search matched **1** of **974953** documents.

A maximum of **1** results are displayed, **15** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.

Then click **Search Again**.

Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **The PowerPC 603 microprocessor: a low-power design for portable applications**
Gary, S.; Dietz, C.; Eno, J.; Gerosa, G.; Sung Park; Sanchez, H.;
Compcon Spring '94, Digest of Papers. , 28 Feb.-4 March 1994
Page(s): 307 -315

[Abstract] [PDF Full-Text (616 KB)] **IEEE CNF**

The PowerPC™ 603 Microprocessor: A Low-Power Design for Portable Applications

Sonya Gary†, Carl Dietz‡, Jim Eno†,
Gianfranco Gerosa†, Sung Park‡, Hector Sanchez†

†Motorola, Incorporated, ‡International Business Machines Corporation
Somerset Design Center, 9737 Great Hills Trail, Austin TX 78759

Abstract

The PowerPC 603™ microprocessor is a low-power implementation of the PowerPC Architecture™. The superscalar organization includes dynamic localized shutdown of execution units to reduce normal-mode power consumption. Three levels of static low-power operation are software programmable for system power management. The 603 PLL (Phase Lock Loop) is capable of generating an internal processor clock at 1X, 2X, 3X or 4X the system clock speed to allow control of system power while maintaining processor performance. Various design features optimize the 603 for both power and performance, creating an ideal microprocessor solution for portable applications.*

1: Introduction

The market for a portable computer is dependent on its performance versus its portability and physical features. The role of the microprocessor in a portable system is important for both of these conflicting demands. The microprocessor must provide the performance and flexibility to support a range of applications while promoting the extension of battery life by minimizing power dissipation during both normal operation and standby.

The PowerPC 603 microprocessor was designed with the performance and low-power requirements of portable systems in mind. The 603 combines an efficient 32-bit implementation of the PowerPC RISC architecture and effective static and dynamic power management features. The 3.3V CMOS design can achieve a peak instruction rate of 3 per cycle and keep power dissipation under 3 watts at 80 MHz.

*In this document, the terms "PowerPC 603 Microprocessor" and "603" are used to denote the second microprocessor from the PowerPC Architecture family. PowerPC, PowerPC Architecture and PowerPC 603 are trademarks of International Business Machines Corporation. SPEC, SPECint92 and SPECfp92 are trademarks of Standard Performance Evaluation Corporation.

2: PowerPC 603 microarchitecture overview

The architectural organization of the 603 focuses on high performance, low cost and power management. Figure 1 shows a high-level block diagram of the 603.

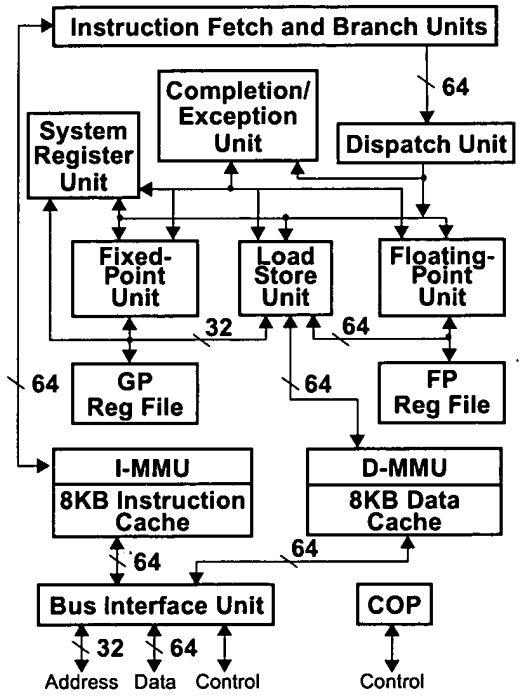

Figure 1: 603 Block Diagram

The instruction fetch unit fetches two instructions per cycle from the instruction cache into a six-entry instruction queue. Branches are folded out of this queue by the branch unit, which can then initiate fetching down either a sequential or target stream. Programmable static branch prediction is used for unresolved branches.

The dispatch unit may issue up to two instructions per cycle to four independent execution units. These units are the fixed-point unit, the floating-point unit, the load/store

unit and the system register unit. With branch folding, a maximum issue rate of three instructions per cycle can be achieved. Execution unit access to general purpose or floating-point registers is handled with rename registers managed by the dispatch unit.

The completion/exception unit tracks all dispatched instructions in a five-entry queue. A maximum of two instructions per cycle are retired in order from this queue. As instructions are retired, architectural register values are committed from the appropriate rename registers. With branch folding, up to three instructions can be completed per cycle.

The 603 contains separate 8KB, 2-way set associative instruction and data caches, each with a 32-byte line size. Both caches are blocking and allow one access per cycle. The data cache supports writeback operation and snoops, while the instruction cache does not support either. Memory management features include dual 64-entry instruction and data translation lookaside buffers, dual 4-entry instruction and data block address translation registers, and sixteen segment registers.

3: Dynamic power management

Dynamic power management is a software-enabled mode on the 603 which turns on power-saving logic for the execution units, caches and memory management units (MMUs) during normal operation. Once the mode is enabled, no other software intervention is necessary.

Dynamic power management logic automatically manipulates clock regenerators to reduce average power consumption. Power is reduced by eliminating clock switching and inhibiting change of registered values. In a static design, if registered values do not change, the logic those values feed does not switch. Clocks to a particular logic block are disabled or enabled on a cycle-by-cycle basis once it is determined whether that block is needed for instruction execution.

3.1: Dynamic power management implementation using clock regenerators

The 603 clock regenerators produce two clocks, C1 and C2, which feed master and slave latches, L1 and L2, respectively. Each clock regenerator features two 'freeze' inputs which are used for dynamic power management control. The assertion of these inputs forces C1 and C2 low. The logic dedicated to support these freeze inputs in the clock regenerators accounts for under 0.3% of the total chip area.

Timing constraints on the assertion of freeze inputs are fairly strict. C1 freeze must be defined by outputs of an L2 latch to ensure freezing the correct C1 pulse and not that of the previous cycle. It must also be early enough to fully block the next C1 pulse. If the assertion of C1 (C2) freeze is late, a small C1 (C2) pulse may be generated. Functionally this is not a problem since the 603 is a fully static design, and latched data is a don't care during the time C1 and C2 are frozen. However, late freeze assertions do not allow maximum power savings. Power is consumed by the extra clock switching and potentially by spurious activity due to data changes. Therefore, it must be determined early in a cycle if a logic block must be clocked at the end of that cycle. This determination produces the C1 freeze. The C1 freeze is fed through a transparent L1 latch to generate the C2 freeze. This is done to ensure a freeze of the correct C2 pulse and not the one of the previous cycle.

Timing constraints on the negation of freeze signals are based on minimum clock pulse width requirements for L1 and L2 and maintaining sufficient usable cycle time. Negation of both C1 and C2 freezes must be early enough to allow at least the minimum pulse width to latch new values so that the 603 functions correctly. The negation of C2 freeze should also be early enough to allow a full C2 pulse so that usable cycle time is not sacrificed by delaying L2 outputs.

C1 freeze and C2 freeze serve to disable or enable a clock regenerator on a cycle-by-cycle basis so that there is no performance sacrifice to the affected logic. Figure 2 shows the required timing for turning the C1 and C2 clocks off and on using the C1 and C2 freeze signals.

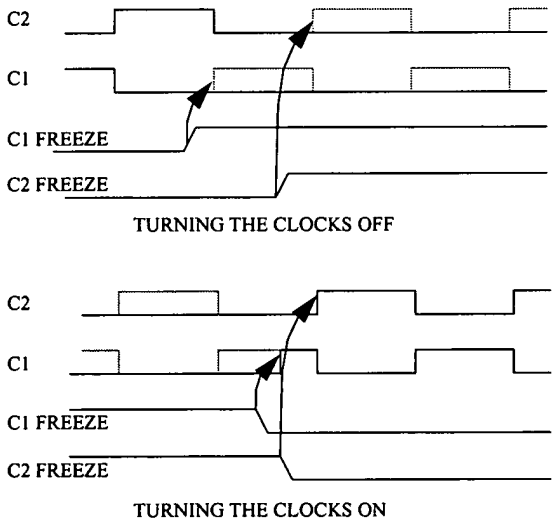

Figure 2: Freeze timing for C1 and C2

3.2: Execution unit dynamic power management

The separation between execution units in the superscalar organization of the 603 allows for independent

dynamic power management in each of the four units. The dynamic power management control in each unit is based on logic already present for instruction dispatch and execution. Only 0.08% of the total chip area of the 603 is attributable to dynamic power management in the execution units.

The 603 execution units employ dynamic power management in a simple fashion. This involves distributed logic for clock freezing based on pipeline stages or the particular instruction dispatched to a unit. The clocks feeding the reservation station for each unit are enabled if a valid instruction is being executed in that unit or if a valid instruction assigned to that unit is present in the instruction dispatch buffer. Whether the instruction is actually dispatched is determined too late in the cycle to be part of the C1 freeze equation. Other stages of the execution unit pipelines are frozen separately based on activity each cycle.

The load/store unit is a good example of the efficiency of this method. The last stage of the store pipe is a completed store queue which holds a store until the cache is available or there is a load dependency. Thus a store may remain in the completed store queue for some time. The 603 may freeze the other stages of the store pipeline despite stores remaining in the completed store queue.

603 execution units also freeze clocks based on the particular instruction dispatched to a unit. The system register unit of the 603, which operates on architectural control registers, is a prime candidate for this method. This unit has only one single-cycle execution stage but manages many registers. Clocks to each register are enabled only when that register is being modified, while clocks to the other registers in this unit remain frozen.

3.2.1: Estimating effectiveness of dynamic power management in the execution units

The effectiveness of using dynamic power management in the execution units varies with the type of code run. For example, the floating-point unit clocks will be frozen continuously if integer-only code is run. If code is scheduled such that all of the units remain busy continuously, few clocks will be frozen.

In evaluating the power saving potential of dynamic power management in the execution units, the percentage idle time of each execution unit was estimated while running different types of code. This was done using an architectural modeling tool, the Basic RISC Architecture Timer (BRAT) [1], which collected statistics while running SPEC92 benchmark traces. Cycles when the dispatch buffer contained an instruction for a unit were not included as idle time for that unit. Though idle time figures indicate the time an entire execution unit may be frozen, they do not account for the time during instruction execution when some portions of that unit may remain frozen.

The BRAT results, shown in Figure 3, illustrate that for some applications, each functional unit may be idle during a large percentage of the run time. Since dynamic power management freezes execution unit clocks during this idle time, it can be effective in reducing the average power consumption of the execution units.

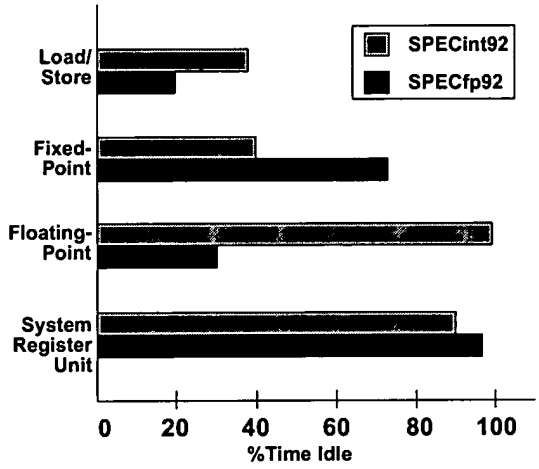

Figure 3: Execution Unit Idle Time for SPEC92

3.3: Cache and MMU dynamic power management

The 603 has blocking caches. This ensures that when a cache miss occurs, all other accesses are held off until miss data is returned from memory. During this time the caches and MMUs are idle and can have their clocks disabled by dynamic power management logic while waiting for data. Depending on the memory latency and the processor to bus clock ratio, this idle time could be many cycles. Blocking caches allowed a straightforward implementation of dynamic power management in the caches and MMUs with little impact on full chip area. The entire dynamic power management logic for the caches and MMUs is approximately 0.20% of the total chip area.

3.3.1: Data cache dynamic power management

Dynamic power management logic freezes all clock regenerators to the data cache while waiting for miss data. This is true for all single beat or burst reads from memory. If a miss in the cache requires a writeback to memory, or castout, the clocks will be frozen after the castout is complete. If a miss in the cache does not require a castout, the clocks are frozen immediately after the miss address is sent to the bus interface unit. The clocks remain frozen until data is returned from memory. Clocks are automati-

cally enabled for each beat of data of a burst read. If there are multiple processor clocks between data beats, the clocks will be frozen between each beat. Figure 4 shows the data cache power management state machine.

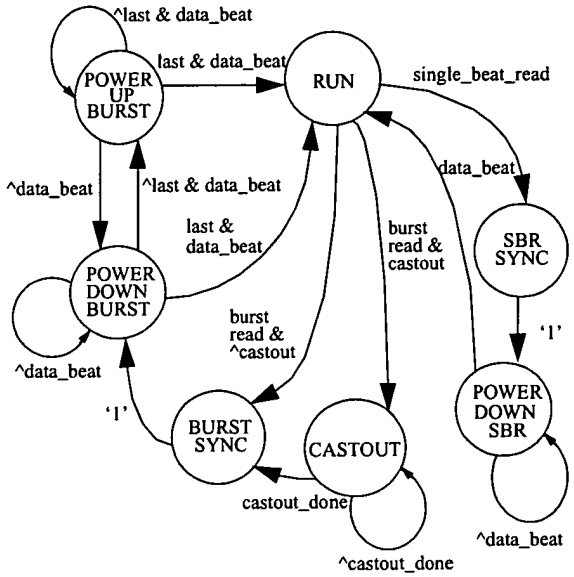

Figure 4: Data Cache Power Management State Machine

3.3.2: Support for bus snooping

The 603 supports bus snooping to maintain memory coherency. If the data cache clocks are frozen due to a miss in the cache, the dynamic power management logic is required to unfreeze the data cache clocks to service snoop requests. If a snoop hits to a modified line in the cache, the data cache clocks will remain enabled for the snoop castout. If the snoop misses or hits to an unmodified line, the data cache clocks can be frozen until it receives a data beat from memory or another snoop request. Figure 5 shows the state machine used to control the unfreezing of clocks for snoop lookups and snoop castouts.

The data cache power management state machine and the snoop power up state machine do not interact. They are totally free-running and operate independently, but together they define the C1 freeze signal for the data cache:

C1_FREEZE = ((POWER_DOWN_SBR & ^data_beat) | (POWER_DOWN_BURST & ^data_beat)) & SNOOP_IDLE & ^snoop_request

This freeze equation allows the data cache to be powered down due to a read miss, then to power up for a snoop request and snoop castout, and then power back down if read data has not been returned from the bus. This approach was found to be much simpler than using one state machine to handle the coordination of cache misses with snoop accesses.

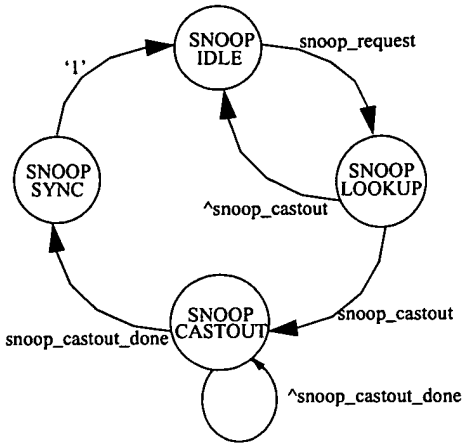

Figure 5: Snoop Power Up State Machine

3.3.3: Instruction cache dynamic power management

The dynamic power management logic for the instruction cache will freeze the clocks on all burst or single beat read misses. The state machine which controls this is identical to the data cache state machine except that the castout section of the state machine is omitted because the instruction cache does not include a modified state. Also, the instruction cache is not snooped so there is no need to power up the instruction cache while waiting for data from the bus.

3.3.4: MMU dynamic power management

Both instruction and data memory management unit clocks are frozen for any burst or single beat read. The MMUs only need to be operational for initial lookups and not during snooping or for data beats from memory, so they are frozen for the entire miss. However, system register unit accesses to the MMUs may occur while the clocks are frozen. If this situation is detected, the dynamic power management logic will unfreeze the clocks until the next power down condition is met. Figure 6 shows the state machine which controls the dynamic power management of the MMUs.

3.3.5: Estimating effectiveness of dynamic power management in the caches and MMUs

To determine the potential power savings of the dynamic power management logic in the caches and MMUs, the percentage of time that the cache would be idle waiting for data from memory was estimated. It was found that the cache would be idle 60% of the time waiting for data, assuming the following: a 6-1-2-1 memory system, a 2:1 processor to bus clock ratio, an 80% cache hit rate, a cache access every cycle, and no bus snooping. Therefore,

dynamic power management logic could freeze the clocks to the caches approximately 60% of the time. The MMUs can be frozen for a greater percentage of the time because they are only used for initial lookups and their clocks do not need to be enabled for each data beat from memory.

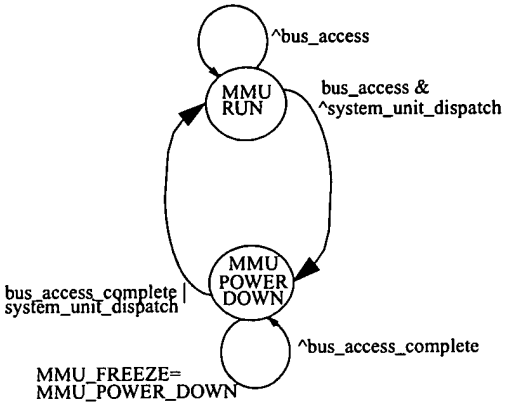

Figure 6: MMU Power Management State Machine

3.4: Dynamic power management results

Table 1 shows total power measured (DC current measured manually) for various applications with 603 silicon using dynamic power management. Power figures include both internal and external power consumed; internal power is measured for internal Vdd supplied to the 603 only, while external power is measured for external Vdd supplied to 603 I/Os plus I/O external pull-ups and terminators. Table 2 shows the percentage decrease in internal power dissipation for the same applications if dynamic power management is used. All of the dynamic power management logic accounts for approximately 0.6% of the total 603 chip area.

Table 1: Dynamic Power Management Results

	Power (W)*				
Freq (MHz)	C linpack	dhrystone	hanoi	heapsort	nsieve
25	0.96	0.85	0.86	0.83	0.90
33	1.16	1.06	1.07	1.02	1.13
50	1.59	1.49	1.48	1.45	1.59
66	1.97	1.86	1.84	1.80	1.98
80	2.21	2.20	2.17	2.12	2.33

*3.3V nominal, room temperature, 80MHz in 2:1 bus mode

Table 2: Percentage Decrease In Internal P wer Dissipati n Using Dynamic Power Management

C linpack	dhrystone	hanoi	heapsort	nsieve
8.5%	14.0%	13.8%	14.2%	16.4%

4: Static power management

The 603 provides three static power management modes, Doze, Nap and Sleep, which are programmable through a hardware implementation register. Static power management allows power management software or an operating system to reduce average power consumption when the 603 is idle for any extended period of time. The names Doze, Nap and Sleep are indicative of the progressive increase in power savings obtainable. Once any one of the static power management modes is enabled, the 603 completes execution of all outstanding instructions and achieves a quiescent state. Once the quiescent state is reached, the 603 disables all clocks to units that are not required to be functional during a particular power management mode. Once clocks are stopped and the processor is in one of the power-saving modes, an external event asserting one of the wake-up signals, such as the external interrupt input pin, will bring the 603 out of that mode. The 603 will resume instruction execution by jumping to the address of the appropriate interrupt vector. This operation is common to all three modes. A more detailed description of the individual modes appears in the following sections.

4.1: Doze mode

Doze mode allows the 603 to maintain cache coherency while in a power-saving mode. The snoop logic and the data cache are kept active to service snoops as they occur on the bus. If a snoop hit occurs, the necessary cache update and a snoop copyback bus operation will occur while the 603 remains in Doze mode. However, if an address parity error occurs during a snoop, or a bus error during a snoop copyback causes a machine check condition, then the 603 will exit the Doze mode and take a machine check interrupt.

Along with the snoop logic, the time base/decrementer logic is kept active in Doze mode. This provides uninterrupted timer functionality during Doze. A decrementer interrupt will cause the 603 to exit Doze mode. Asserting any of the pins INT_, SMI_, MCP_, SRESET_ or HRESET_ will also cause the 603 to exit Doze mode in ten system clocks (SYSCLKs) or less. Using a 1:1 proces-

sor to bus clock ratio, this is the worst case logic delay from the external pin assertion to all clocks being enabled.

4.2: Nap mode

In Nap mode, all logic is disabled except the time base/decrementer logic. Since cache coherency cannot be maintained in Nap mode, the system program must flush the data cache before entering Nap mode if system memory is expected to be altered while in Nap mode. Otherwise, a data cache flush is not necessary since the 603 keeps the cache contents unaltered while in Nap mode. A pair of handshake signals, QREQ_ and QACK_, are provided for the system to allow the 603 to go into Nap only when cache coherency will no longer be a problem. When the 603 is ready to enter the Nap mode, QREQ_ is asserted. The system logic responds with an active QACK_ to allow the 603 to proceed with entering Nap mode.

Compared to Doze mode, further power savings are achieved with the data cache and bus snooping logic disabled. Furthermore, the receivers of most input and bidirectional pins are disabled for added power savings. Outputs maintain their normal idle state. A decrementer interrupt or assertion of any one of the pins mentioned previously with regard to Doze mode will wake the 603 from Nap mode. As in the Doze mode, the wake-up latency is ten SYSCLKs or less.

4.3: Sleep mode

Sleep mode allows a maximum power savings by disabling the clocks to all units. The same QREQ_/QACK_ handshake protocol exists for Sleep mode as for Nap. Unlike the Doze and Nap modes, more power savings can be achieved in Sleep mode by disabling the PLL and SYSCLK. In Doze and Nap modes, the SYSCLK and PLL configurations must remain in the same state as they were prior to the commencement of the power-saving mode. Sleep mode provides the ability to dynamically manage system power by allowing system logic to disable the PLL or SYSCLK, change the SYSCLK frequency, or change the processor to bus clock ratio through the PLL configuration pins.

Since Sleep mode does not automatically disable the PLL, the system logic can implement several different levels of power savings depending on the wake-up response time requirements. For example, if a quick wake-up from sleep is required, then by leaving the SYSCLK input and the PLL configuration unaltered, the 603 will wake from Sleep within ten SYSCLK cycles or less. However, if maximum power savings is a requirement, then the PLL and SYSCLK may be disabled completely, reducing power dissipation to leakage levels. However, when coming out of Sleep mode, a maximum of 100 usec (10 usec typical) is required for the PLL to relock to the new SYSCLK frequency. After this relock time, any of the external pins mentioned in the Doze section may be applied to wake-up from Sleep.

4.4: Static power management results

Table 3 shows total power measured with 603 silicon using the static power management modes. Sleep figures are listed for configurations with the PLL and SYSCLK enabled, with the PLL disabled and SYSCLK enabled, and with both the PLL and SYSCLK disabled.

Table 3: Static Power Management Results

Freq. (MHz)	Power (mW)*				
	Doze	Nap	Sleep		
			PLL on CLK on	PLL off CLK on	PLL off CLK off
25	133.2	49.4	38.8	12.8	4.7
33	168.0	62.0	47.8	13.5	4.7
50	241.7	88.8	66.2	15.1	4.7
66	307.1	113.0	88.5	17.7	4.7
80	366.1	135.1	105.5	19.3	4.7

*3.3V nominal, room temperature, 80MHz in 2:1 bus mode

5: Other low-power design features

Several other areas of the 603 were targeted for low-power design. These include the caches and memory management units, the phase lock loop, clock distribution and clock regenerators, standard cell and datapath libraries and the bus interface unit.

5.1: Cache and memory management design

The data portion of each cache is divided into eight 1KB subarrays, which were specifically designed for low-power operation. Each subarray holds one word of the cache line for both sets of data. This organization was chosen to minimize power: since each cache employs double-word datapaths, there is never a need to access more than two subarrays during any given cycle. If only one word of data needs to be read or written, only one subarray is accessed.

Each subarray is held in a constant state of precharge if

the array is not being accessed. That is, each bitline and output driver of each subarray is constantly being precharged when the subarray is idle. To save power, the precharge signals are not clocked and only turn off while the subarray is being accessed. The precharge of the output drivers is disabled only if the access is a read access.

Special attention was given to the timing of the precharge signals and the word lines in each subarray to guarantee that no bitcell is ever enabled while the bitline precharge devices are enabled. To further save power, each subarray employs pulsed word lines to eliminate unnecessary bitline discharge once the sense amplifiers have been strobed. The sense amplifiers within each subarray function as data latches once they have been strobed, eliminating the need for clocked output data latches. For speed reasons, the sense amplifiers for both sets of data within each subarray are strobed for load hits. For castouts, however, only the sense amplifiers for the data set being cast out are strobed. Each subarray contains a way-select mux, allowing both sets of data to share the same output drivers.

The tag portion of each cache consists of a single array. To save power, each tag array is precharged when idle. Due to the speed requirements of supporting a read-modify-write operation each cycle, the precharge signals within each tag array are clocked. Like the data subarrays, each tag array employs pulsed wordlines and strobed sense amplifiers. Unlike the data subarrays, however, each tag array's wordline pulsing logic is designed to limit the bitline discharge to approximately one volt, regardless of frequency. Limiting bitline discharge to one volt saves power at lower frequencies, where one of each pair of bitlines would discharge completely.

The 603's separate instruction and data MMUs each consist of three arrays: a sixteen-entry Segment Register array, a 4-entry Block-Address-Translation (BAT) array, and a 64-entry, 2-way set associative Translation Lookaside Buffer (TLB) array. Like the cache tag arrays, each of these arrays is precharged when idle, but the precharge signal is clocked. Each of these arrays is so small and fast that sense amplifiers are not required and thus pulsing the word lines of these arrays is unnecessary. This provides a further savings in DC power. The TLB array contains a way-select mux, allowing both sets of TLB data to share the same output drivers.

5.2: Phase Lock Loop

The 603 integrates a synthesizing analog PLL which maintains proper edge alignment between the 603's internal clock and SYSCLK. In addition, the 603 PLL can be configured during power up for a processor clock frequency that is 1X, 2X, 3X, or 4X the bus clock frequency. This allows reduced system power for a given level of processor performance. The PLL is designed to lock to a wide range of SYSCLK frequencies from 16 MHz up to 66 MHz. If lower frequencies are desired, the PLL can be bypassed altogether. During Sleep mode, all analog circuits used in the PLL design are completely shut down in order to attain milliwatt performance.

5.3: Clock distribution

A passive clock distribution network was chosen over a set of distributed clock buffers in order to minimize active power. An HTREE style distribution network (see Figure 7) using metal3 and metal4 produced the best rise/fall time performance while maintaining a low total routing capacitance. In addition, skew was minimized among all branches. Grid-like layouts were discounted due to their high routing capacitance. Global clock skew was kept under +/- 100 pS with this scheme; rise and fall times were under 1.2 nS while average power due to the HTREE driver and the HTREE itself was kept under 100 mW at 80 MHz.

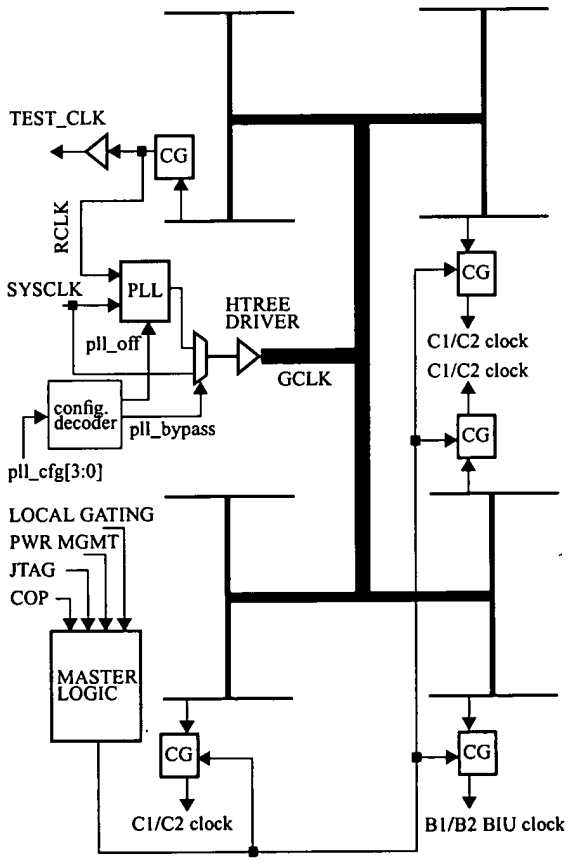

Figure 7: HTREE Clock Distribution Network

5.4: Clock regenerator design

The output drivers of all 603 clock regenerators are designed with enough granularity to cover a wide capacitive load range from 0.2 - 2.2 pF. As a result, lower average power is obtained as compared to other schemes which use 'dummy' loads in order to maintain constant clock latencies.

5.5: Standard cell and datapath libraries

Static logic design is used throughout the standard cell and datapath element libraries where the (speed X power) product is optimized. Special care was taken during logic synthesis in repowering standard cells for greater drive capability. Likewise, datapath elements were only repowered in order to meet certain timing constraints.

5.6: Bus Interface Unit

In normal operation, the bus interface unit (BIU) reduces power consumption by turning off the 64 bit data bus receivers when the 603 is not reading from the bus. In this way power is saved whenever the data bus is used by other masters for operations such as DMA transfers or when the 603 is performing write operations. Dynamic power management logic is not implemented in the BIU due to bus snooping requirements. Static power management modes do affect the BIU. In Nap and Sleep modes, clocks to the BIU are disabled along with any power-consuming circuitry, such as a voltage reference generator. Also, when waking up from Nap or Sleep, the BIU resumes normal operation only after the bus clock and processor clock have been synchronized.

6: CMOS Technology

The 603 is fabricated with a 3.3 volt 0.5 micron CMOS technology using four layers of metal. Split instruction and data caches account for over half of the 1.6 million transistors used within an 85 mm^2 die. The design is fully static and LSSD compliant. 603 physical characteristics are summarized in Table 4.

7: 603 performance

The 603 design is optimized for both power and performance. Table 5 provides estimated SPEC92 performance for an 80MHz 603-based system using a 2:1 processor to bus clock ratio and a second level external cache [2].

8: Conclusion

The PowerPC 603 microprocessor is a low-power implementation of the PowerPC Architecture with power management capability and competitive performance. The area-conscious design also makes the 603 a cost-effective alternative. The 603 combines all of the necessary features to make it the ideal microprocessor solution for portable applications.

Table 4: 603 Physical Characteristics

Technology:	0.5 um Nwell CMOS, four-layer metal
Die Size:	7.39 mm X 11.47 mm
Transistors:	1.6 million
Memory:	split 8KB caches
Bus:	split 32b address, 64b data, 16 - 66 MHz
Voltage:	3.3 V nominal
Power:	3.0 Watts Max @ 80 MHz
Package:	240 pin wire-bond CQFP, C4 CQFP
Frequency:	16 - 80 MHz
Processor to bus clock ratios: 1:1, 2:1, 3:1, 4:1	
I/Os:	165 signal, CMOS/TTL, 5V tolerant

Table 5: 603 Performance Estimate

Frequency	SPECint92	SPECfp92
80MHz	75*	85*

*estimated based on simulated results

Acknowledgments

The authors gratefully acknowledge the 603 design team for their support in the design, implementation and verification of the power-related features of this part. In particular, Peter Ippolito provided the clock regenerator design which made the implementation of the power management features possible. Also, the support and promotion of the 603 as a power-conscious microprocessor by the 603 project managers, Arturo Arizpe and Jim Kahle, was invaluable.

References

[1] Poursepanj, A. et al., "The PowerPC™ 603 Microprocessor: Performance Analysis and Design Trade-Offs," *Proceedings of COMPCON 1994*, February 1994.

[2] Burgess, B., et al., "The PowerPC™ 603 Microprocessor: A High Performance, Low Power, Superscalar RISC Microprocessor," *Proceedings of COMPCON 1994*, February 1994.

[3] Arizpe, A., and Kahle, J., "The Future Direction of PowerPC™," The Sixth Annual Microprocessor Forum, October 18, 1993.